

1
2 **CLAIMS**

3 1. A method, comprising:
4 receiving an input that includes hierarchical data;
5 traversing a filter tree according to segments of the hierarchical data to
6 locate one or more matching nodes that correspond to the hierarchical data;
7 comparing at least a portion of the input to one or more filters associated
8 with the matching nodes; and
9 executing instructions associated with one or more filters satisfied by the
10 input.

11
12 2. The method as recited in claim 1, further comprising applying a
13 precedence rule to the matching nodes, wherein the comparing step is performed
14 on a subset of the matching nodes that is determined by the precedence rule.

15
16 3. The method as recited in claim 1, wherein a node may be associated
17 with instructions but no filter, in which case the input is deemed to match a filter
18 for the node, thereby resulting in execution of the instructions.

19
20 4. The method as recited in claim 1, wherein the hierarchical data
21 further comprises a path that identifies a location in a hierarchical system.

22
23 5. The method as recited in claim 1, wherein:
24 a root node of the tree corresponds to a first segment of the hierarchical
25 data;

1 a child node of the root node corresponds to a second segment of the
2 hierarchical data; and

3 a bottom-level node of the tree corresponds to a last segment of the
4 hierarchical data.

5
6 6. The method as recited in claim 1, wherein each node of the filter tree
7 references zero or more filters.

8
9 7. The method as recited in claim 1, wherein the hierarchical data
10 further comprises a destination path identified by a segment string.

11
12 8. The method as recited in claim 7, wherein the input further comprises
13 message data that is transmitted to a location identified by the destination path if
14 the input satisfies a filter that is referenced by a filter tree node associated with the
15 destination path.

16
17 9. The method as recited in claim 1, further comprising:
18 identifying the hierarchical data contained in the input; and
19 parsing the hierarchical data into segments for use with matching.

20
21 10. The method as recited in claim 1, wherein the traversing step further
22 comprises:

23 comparing a first segment of the hierarchical data with a first node in a
24 filter tree level that corresponds with a position of the first segment in the
25 hierarchical data;

1 if the first segment does not match the first node, determining that the input
2 does not match the first node;

3 if the first segment matches the first node and there is a subsequent second
4 segment in the input, comparing the subsequent second segment to one or more
5 second nodes in the filter tree that are subordinate to the first node; and

6 if the first segment matches the first node and there is not a subsequent
7 second segment in the input, determining that the input matches the first node.

8
9 **11.** A filter tree data structure stored on one or more computer-readable
10 media, comprising:

11 a first level having a root node that corresponds to an initial segment of
12 hierarchical data;

13 at least one intermediate level having at least an intermediate node that
14 corresponds to an intermediate segment of the hierarchical data, the intermediate
15 node being subordinate to the root node;

16 a bottom level having at least a bottom level node that corresponds to a
17 final segment of the hierarchical data, the bottom level node being subordinate to
18 an intermediate node; and

19 wherein at least one node is an active node that references an instruction set
20 that is executed when an input is received that includes the segment corresponding
21 to the active node and segments corresponding to all nodes superior to the active
22 node.

23
24 **12.** The filter tree as recited in claim 11, wherein:

25 the active node further comprises a reference to a filter; and

1 the instruction set is executed only if the input satisfies the filter.

2
3 13. The filter tree as recited in claim 11, wherein the instruction set if
4 executed only if the active node is a bottom level node.

5
6 14. A system, comprising:

7 memory;

8 a filter tree stored in the memory, at least one node of the filter tree
9 referencing a filter;

10 a message input module configured to receive a message that includes a
11 path having one or more segments;

12 a primary matching module configured to locate one or more filter tree
13 nodes that match one or more of the segments;

14 a secondary matching module configured to identify any filters associated
15 with the one or more matching filter tree nodes and to compare the message
16 against the filters to determine if the message satisfies any of the filters; and

17 a message processing module configured to execute instructions associated
18 with any filter that is satisfied by the message.

19
20 15. The system as recited in claim 14, wherein the secondary matching
21 module is further configured to apply a precedence rule to the one or more
22 matching filter tree nodes to derive a subset of the matching filter tree nodes and
23 to identify only the filters associated with the subset of matching filter tree nodes.

24
25 16. The system as recited in claim 14, wherein the filter referenced by

1 the at least one node further comprises a null filter that is deemed to be satisfied by
2 any input message compared thereto, thereby resulting in the execution of
3 instructions associated with the node referencing the null filter.
4

5 17. The system as recited in claim 14, wherein the message path is of a
6 hierarchical nature and successive path segments correspond to successively
7 subordinate levels of the filter tree.
8

9 18. The system as recited in claim 14, wherein the primary matching
10 module is further configured to:

11 identify one or more path segments included in the message;
12 locate filter tree nodes associated with each path segment; and
13 determine that each node located matches the message.
14

15 19. The system as recited in claim 14, wherein the primary matching
16 module is further configured to:

17 identify one or more path segments included in the message;
18 locate filter tree nodes associated with each path segment; and
19 determine that a node associated with a final path segment is the only node
20 that matches the message.
21

22 20. The system as recited in claim 14, wherein the primary matching
23 module is further configured to:

24 identify one or more path segments included in the message;
25 locate filter tree nodes associated with each path segment; and

1 determine that a node associated with an initial path segment is the only
2 node that matches the message.

3
4 **21.** A method, comprising:

5 receiving a data transmission requesting to add an entry to memory that
6 stores multiple filters, the data transmission including at least a new filter, a
7 segmented path identifying hierarchical reference data associated with the new
8 filter, and at least one data item associated with the new filter;

9 traversing a hierarchical data structure stored in the memory that is used to
10 reference each of the multiple filters to determine if an existing location in the data
11 structure matches the segment path included in the data transmission;

12 if an existing location is identified, adding a reference to the new filter to
13 the existing location; and

14 if an existing location is not found, creating a new location in the data
15 structure, the new location being determined according to the hierarchical
16 reference data and storing a reference to the new filter at the new location.

17
18 **22.** The method as recited in claim 21, further comprising storing a
19 reference to the data item at the new location so that an input matching the new
20 location can access the data item.

1 **23.** The method as recited in claim 21, wherein:

2 each of the multiple filters stored in the memory is associated with
3 hierarchical reference data; and

4 the memory includes a filter tree data structure to reference the multiple
5 filters according to the hierarchical reference data, each of the multiple filters
6 being associated with a node in the filter tree data structure.

7
8 **24.** The method as recited in claim 23, wherein:

9 each segment of the segmented path corresponds to a node in the filter tree
10 data structure so that a node associated with a filter corresponds with a final
11 segment of the segmented path; and

12 each of one or more superior nodes to the node associated with the filter
13 corresponds with a preceding segment of the segmented path.

14
15 **25.** One or more computer-readable media containing computer-
16 executable instructions that, when executed on a computer, perform the following
17 steps:

18 receiving a data transmission that includes at least a message and a
19 destination path that is structured according to a hierarchical protocol;

20 parsing the destination path into constituent segments;

21 comparing the destination path segments to a hierarchical data structure that
22 is arranged according to the hierarchical protocol to determine one or more
23 matching locations in the data structure that correspond with the destination path;

24 identifying one or more filters associated with the one or more matching
25 locations;

1 testing the message against one or more filters referenced by the one or
2 more matching locations; and

3 for each filter satisfied by the message, executing one or more instructions
4 associated with the filter.

5
6 **26.** The one or more computer-readable media as recited in claim 25,
7 wherein the hierarchical protocol is eXtensible Markup Language (XML)
8 protocol.

9
10 **27.** The one or more computer-readable media as recited in claim 25,
11 wherein the one or more filters are defined using Xpath.

12
13 **28.** The one or more computer-readable media as recited in claim 25,
14 wherein the one or more instructions associated with the filter further comprise a
15 destination to which at least a portion of the message is transmitted.

16
17 **29.** The one or more computer-readable media as recited in claim 25,
18 wherein the destination path is a network address.